UNITED STATES PATENT APPLICATION

FOR

METHOD AND SYSTEM FOR PROVIDING AN OPEN AND INTEROPERABLE SAML SESSION

Inventors:

CHENG, Qingwen
BHATNAGAR, Bhavna
XU, Hong
SUN, Wei
LUO, Ping
BHAT, Shivaram
RANGANATHAN, Aravindan

Prepared by:
WAGNER, MURABITO & HAO, LLP
Two North Market Street
Third Floor
San Jose, California 95113

SUN-P8959

METHOD AND SYSTEM FOR PROVIDING AN OPEN AND INTEROPERABLE SYSTEM

FIELD OF THE INVENTION

Embodiments of the present invention relate to network communication systems, and more particularly to an open and interoperable SAML session on a network.

BACKGROUND OF THE INVENTION

Referring to Figure 1, a block diagram of a network according to the conventional art is shown. As depicted in Figure 1, a plurality of organizations 105-120 are communicatively coupled by one or more communication channels 190, 195, such as the internet or an extranet. Each organization 105-120 typically comprises a plurality of client devices 125-165 communicatively coupled to one or more servers 170-182. The servers 170-182 provide one or more resources, such as execution of applications and/or storage of information.

In the conventional art, the client 125 logs-on to a particular entity's server 170, wherein the user provides a username, password and/or the like. Based upon the username, password and the like, the server 170 authenticates the client 125 and determines the client's 125 authorization to access particular resources.

If the client 125 then tries to access resource on another server 172, establishing authentication and utilizing resources is problematic. The other servers 172-182 do not know

that the client device 125 has been authenticated by a particular server 170. Furthermore, each organization 105-120 and/or server 170-182 may have a different login script, may require a different protocol, may utilize different information, may store the same information in differing structures, formats and/or the like. Therefore, the client typically has to sign onto each server 170-182 separately.

For example, a user may wish to access resources on various organizations 105-115 in the performance of their work, such as using the internet 190 to make travel arrangements. The user first logs-on to the company's (e.g., Widget Corp.) network server 170 utilizing a client device 125. The user may manually or via a script, enter their username (e.g, jdoe) and password in order to logon to the network server 170. The network server 170 provides an internet portal.

The user may then navigate using a browser to the website of an airline 115. The user will likely be required to enter their name, address and the like to book a flight. The user's address may be stored as a single field (e.g., address) in a record corresponding to the user's reservation. Similarly, the user may then navigate to a car rental agency 110 to reserve a rental car. Once again, the user may be requested to enter their name and address to reserve the car. The user's address may be stored as a plurality of fields (e.g., street address, city, state, zip code) in the record corresponding to the user's reservation. Similarly, the user may also navigate to a website of a hotel chain 120 to reserve a room. The user may be requested to enter a username (e.g, janed), password and the like to reserve the room using a corporate account.

Heterogeneous systems require various information. Some information may be common to each, while other information may be unique to one or more systems. Each system may also store the same information in various different formats. Thus, the interoperability of the various systems is problematic. Furthermore, the need to logon to each entity's server 170-172 and re-enter the same information (e.g., address, username) reduces the users satisfaction and productivity.

The need to logon multiple times is not limited to moving between multiple entities 105-120. For example, the user may login to their employer's network server 170 to access the finance server 180. The user may again be required to enter a username, password and the like in order to enter expenses, such as meals, entertainment and gas, incurred during their business travel. Along with entering a username, password and the like, the user may be required to enter their social security number, which is utilized to limit authorization to resources on the finance server 180. The financial server 180 may save the social security number in a field entitled "ssn." The user may then wish to check their retirement account. Once again the user may be required to provide a username, password and the like to access the payroll server 182 in order to check their retirement account. Along with enter a username, password and the like, the user may be required to enter their social security number, which the server utilizes to limit authorization to utilize resources on the payroll server 182. The financial server 180 may save the social security number in a field entitled "social security." Accordingly if different fields contain the same information, the interoperability of the systems may be limited even though the fields are substantially the same.

The Security Assertion Markup Language (SAML) specification is intended to provide a solution allowing single sign-on for secure authentication and authorization. SAML is an eXtensible Markup Language (XML) standard designed for business-to-business (B2B) and business-to-consumer (B2C) transactions.

Referring now to Figure 2A, an exemplary SAML request/assertion according to the conventional art is shown. As depicted in Figure 2A, SAML requests and assertions 210 are transmitted within a SOAP envelope 215 via HTTP 220.

Referring now to Figure 2B, an exemplary SAML data packet according to the conventional art is shown. As depicted in Figure 2B, the data packet comprises an HTTP header 250, a SOAP header 255 and a SAML payload 260. A request or assertion is encoded into the SAML payload 260. A SOAP header 255 is then generated and attached to the SAML payload 260. An HTTP header 250 is then generated and attached to the SOAP header 255 and SAML payload 260. The SAML payload containing a request or assertion may comprise an issuer identifier, an assertion identifier, an optional subject, an optional advice, a condition, an audience restriction, a target restriction, an application specific condition and the like.

Upon receipt, the HTTP header 250 is processed to provide routing and flow control. The SOAP header 255 is then processed to provide information concerning the content of the payload and how to process it. The SAML payload 260 may then be processed to provide security information.

Security Assertion Markup Language (SAML) for single sing-on functionality is intended to allow users to authenticate themselves in one domain and use the resources in another domain without re-authenticating themselves. SAML is intended to be an open and interoperable design for web-based single sign-on service functionality. However, differences in the organization and utilization of information limit the application of SAML in a heterogeneous network.

SUMMARY OF THE INVENTION

Embodiments of the present invention provide an open and interoperable single sign-on session in a heterogeneous communication network. Embodiments of the present invention advantageously enable interoperation of disparate security service systems. Embodiments of the present invention comprise mapping elements of a SAML request or assertion that related to the content and organization of information utilized by the present entity to the corresponding content and organization of information utilized by a partner entity.

Embodiments provide a method and system of configuring an open and interoperable single sign-on session. An administration module provides for receipt of an entity identifier of a partner entity, an account mapping between the partner entity and the present entity, an attribute mapping between the partner entity and the present entity, a site attribute list between the partner entity and the present entity, an action mapping between (e.g., authorization decision) the partner entity and the present entity, and/or the like. The entity identifier, account mapping, attribute mapping, site attribute list, action mapping and the like for each partner entity is stored in a trusted partner list accessable to the present entity.

Embodiments are directed to a method and system of providing an open and interoperation single sign-on session. Upon receipt of a SAML request or assertion containing an entity identifier, the entity identifier is looked-up in a trusted partner list of the present entity. A record containing a matching entity identifier provides the applicable

account mapping, attribute mapping, site attribute list, and/or action mapping. The one or more mappings are then utilized to process or generate the SAML request or assertion.

Embodiments of the present invention provide a unique SAML configuration defining the mapping between heterogeneous partner sites. An account mapping defines how a subject between two entities are mapped. An attribute mapping defines how an attribute is mapped from a first entity to a second entity. A site attribute list defines an attribute that needs to be returned by said first entity to said second entity. An action mapping defines how an authorization decision of said first entity is mapped to and authorization decision of said second entity

BRIEF DESCRIPTION OF THE DRAWINGS

The present invention is illustrated by way of example and not by way of limitation, in the figures of the accompanying drawings and in which like reference numerals refer to similar elements and in which:

Figure 1 shows a block diagram of a network according to the conventional art.

Figure 2A shows an exemplary SAML request/assertion according to the conventional art.

Figure 2B shows an exemplary SAML data packet according to the conventional art.

Figure 3 shows a flow diagram of a computer-implemented method of configuring an interoperable single sign-on system, in accordance with one embodiment of the present invention.

Figure 4 shows an exemplary trusted partner list, in accordance with one embodiment of the present invention.

Figure 5 shows a block diagram of a system providing for configuring an interoperable single sign-on system, in accordance with one embodiment of the present invention.

Figure 6 shows a computer-implemented method of providing an interoperable single sign-on system, in accordance with one embodiment of the present invention.

Figure 7 shows a block diagram of a system providing an interoperable single sign-on session, in accordance with one embodiment of the present invention.

DETAILED DESCRIPTION OF THE INVENTION

Reference will now be made in detail to the embodiments of the invention, examples of which are illustrated in the accompanying drawings. While the invention will be described in conjunction with these embodiments, it will be understood that they are not intended to limit the invention to these embodiments. On the contrary, the invention is intended to cover alternatives, modifications and equivalents, which may be included within the spirit and scope of the invention as defined by the appended claims. Furthermore, in the following detailed description of the present invention, numerous specific details are set forth in order to provide a thorough understanding of the present invention. However, it is understood that the present invention may be practiced without these specific details. In other instances, well-known methods, procedures, components, and circuits have not been described in detail as not to unnecessarily obscure aspects of the present invention.

Referring now to Figure 3, a flow diagram of a computer-implemented method of configuring an interoperable single sign-on system, in accordance with one embodiment of the present invention, is shown. The method of the present embodiment may be realized as a series of instructions (e.g., code) and information (e.g., data) that reside on a computer-readable medium, such as computer memory, and are executed and manipulated by a processor. When executed, the instructions cause the processor to implement the process of configuring an interoperable single sign-on system. As depicted in Figure 3, the method comprises receiving an entity identifier (e.g., server A), a certificate, a network address (e.g., internet protocol address) and/or the like of a first entity by a second entity, at step 310.

The entity identifier, certificate, network address and/or the like uniquely identifies the particular entity. The method further comprises receiving an account mapping between the first entity and the second entity by the second entity, at step 320. The account mapping defines how the subject between the first entity (e.g., local site) and the second entity (e.g., partner site) are mapped to each other. In an exemplary implementation, the account mapping comprises a java class which provides an interface that is implemented to map a partner account of the first entity to a user account of the second entity.

The method may optionally comprise receiving an attribute mapping, a site attribute list, action mapping and/or the like, between the first entity and the second entity, by the second entity. The attribute mapping defines how an attribute and/or an attribute namespace between the first entity and the second entity are mapped. The site attribute list defines what attributes need to be exchanged between the first entity and second entity. The action mapping defines what authorization decision information are exchanged between the first entity and the second entity.

The entity identifier, client certificate, network address and/or the like of the first entity is stored as one or more fields of a corresponding record in a partner list accessable to a second entity, at step 330. The account mapping between the first entity and the second entity is stored as another field of the corresponding record in the partner list accessable to the second affiliated entity. Optionally, the attribute mapping, the site attribute list, action mapping and/or the like of the first entity to the second entity may be stored as additional fields of the corresponding record in the partner list accessible to the second affiliated entity.

The method further comprises receiving an entity identifier, a client certificate, network address and/or the like of the second entity by the first entity, at step 340. The method further comprises receiving an account mapping between the second entity and the first entity by the first entity, at step 350. The method may optional further comprise receiving an attribute mapping, a site attribute list, action mapping and/or the like, between the second entity and the first entity, by the first entity.

The entity identifier, certificate, network address and/or the like of the second affiliated entity is stored as one or more fields of a corresponding record in a partner list accessable to the first affiliated entity, at step 360. The account mapping, between the second entity and the first entity is stored as another field of the corresponding record in the partner list accessable to the first affiliated entity. Optionally, the attribute mapping, the site attribute list, action mapping and/or the like of the second entity to the first entity may be stored as additional fields of the corresponding record in the partner list accessible to the first entity.

The method may be repeated for each pair of entities in a network. Thus, the partner list of each entity may comprise a record, including an entity identifier and an account mapping, for each partner entity. Each record in an entity's partner list may further include a corresponding client certificate, network address, attribute mapping, site attribute list, account mapping, and/or the like.

Referring now to Figure 4, an exemplary trusted partner list, in accordance with one embodiment of the present invention, is shown. As depicted in Figure 4, the trusted partner list of a first entity comprises a plurality of records 410. In one implementation, each record 410 comprises an identifier of a second entity 420, and an account mapping between the first entity and the second entity 450. Each record 410 may further comprise an a certificate 430, network address 440 and/or the like corresponding to the second entity, an attribute mapping 460, a site attribute list 470, and action mapping 480 and/or the like, between the first entity and the second entity.

For example, the partner list for server A may comprise entity identifiers for servers B and C, and account mappings of A to B and A to C. Additionally, the partner list for server A may further comprise certificates of B and C, network address (e.g., internet protocol addresses) of B and C, attribute mappings of A to B and A to C, site attribute lists of A to B and A to C, and action mappings of A to B and A to C.

Referring now to Figure 5, a block diagram of a system providing for configuring an interoperable single sign-on system, in accordance with one embodiment of the present invention, is shown. As depicted in Figure 5, each entity (e.g., server A and B) 510, 520 comprises an administration module 530, 540 and a partner list 550, 560, respectively. The administration module 530 of the first entity 510 receives an entity identifier and/or the like of a second entity 520 and stores it in a first field of a record in the partner list 550 of the first entity 510. The administration module 530 of the first entity 510 also receives a mapping between an account of the second entity 520 and an account of the first entity 510.

The administration module 530 of the first entity 510 stores the account mapping in a second field of the corresponding record in the partner list 550 of the first entity 510.

Optionally, the administration module 530 of the first entity 510 may also receive a client certificate and/or network address of the second entity 520. The administration module 530 of the first entity 510 may also receive an attribute mapping, a site attribute list, an action mapping and/or the like between the second entity and the first entity. The administration module 530 of the first entity 510 stores the certificate, network address, attribute mapping, site attribute list, action mapping and/or the like in a plurality of additional fields of the corresponding record in the partner list 550 of the first entity 510.

Similarly, the administration module 540 of the second entity 520 receives an entity identifier and/or the like of the first entity 510 and stores it in a first field of a record in a partner list 560 of the second entity 520. The administration module 540 of the second entity 520 also receives a mapping between an account of the first entity 510 and an account of the second entity 520. The administration module 540 of the second entity 520 stores the account mapping in a second field of the corresponding record in the partner list 560 of the second entity 520.

Optionally, the administration module 540 of the second entity 520 may also receive a client certificate and/or network address of the first entity 510. The administration module 540 of the second entity 520 may also receive an attribute mapping, a site attribute list, an action mapping and/or the like between the first entity 510 and the second entity 520. The

administration module 540 of the second entity 520 stores the certificate, network address, attribute mapping, site attribute list, action mapping and/or the like in a plurality of additional fields of the corresponding record in the trusted partner list 560 of the second entity 520.

The client certificate, network address, account mapping, attribute mapping, site attribute list, action mapping and/or the like may be provided to the first and second entities by an administrator or the like. In one implementation, the mappings are implemented as a java class. The account mapping defines how the subject between two sites (e.g., the first and second entities 510, 520) are mapped to each other. In one implementation, the account mapping between a partner source identifier and the implementation class are configured at the partner URLs field in a SAML service. The attribute mapping defines how attributes between two sites are mapped. In one implementation, the attribute mapping between the partner source identifier and the implementation class are configured at the partner site field in the SAML service. The site attribute list defines the list of attributes which will be returned as attribute statement elements as part of an assertion. In one implementation, the site attribute list between the partner source identifier and the implementation class are configured at the partner URLs field in the SAML service. The action mapping defines how the second entity's authorization decision is mapped to a policy decision of the first entity's policy decision. In one implementation, the action mapping between the partner source identifier and the implementation class are configured at the partner site field in the SAML service. Accordingly, the mappings may advantageously comprise straight mappings or mappings of any complexity.

Referring now to Figure 6, a computer-implemented method of providing an interoperable single sign-on system, in accordance with one embodiment of the present invention, is shown. The method of the present embodiment may be realized as a series of instructions (e.g., code) and information (e.g., data) that reside on a computer-readable medium, such as computer memory, and are executed and manipulated by a processor. When executed, the instructions cause the processor to implement the functionality, processes and/or benefits of an interoperable single sign-on system. As depicted in Figure 6, the method begins with receipt by a first entity of a SAML request from a second entity, at step 610. The SAML request contains an entity identifier, certificate, network address and/or the like. At step 620, the entity identifier, certificate, network address and/or the like contained in the SAML request is looked-up in a partner list of the first entity. An account mapping, attribute mapping, site attribute list, action mapping and/or the like is obtained from a record in the partner list containing the matching entity identifier, certificate, network address and/or the like.

Thereafter, the SAML request may be processed according to the account mapping, attribute mapping, site attribute list, action mapping and/or the like at step 630. In an exemplary implementation, a SAML request received at a local entity from a partner entity may indicate that a client (e.g., client 1A) is logged-on and has a username (e.g., jdoe). Therefore, the local entity (server B) may process the SAML request utilizing the account mapping provided in its partner list. The account mapping for example, may map the username of the partner entity to the username on the local entity (e.g., janed). The

exemplary account mapping may comprise "if user (server A) = jdoe, then user (server B) = janed."

In another exemplary implementation, a SAML request received at a local entity from a partner entity may indicate the client is logged-on and has an address (e.g., 101 First Ave., Anytown, California 12121), wherein the address is passed as four fields (e.g, street, city, state, and zip). Therefore, the local entity may process the SAML request utilizing the attribute mapping provided in its partner list. The attribute mapping for example, may map the address of the partner entity, wherein the address is utilized as four fields, to the address on the local entity, wherein the address is utilized as a single field. In so doing the mapping may cause the four address fields to be concatenated to form a single address field.

In yet another exemplary implementation, a SAML request received at the local entity may indicate that the client is logged-on and only has permission to access records in a system (e.g., financial) corresponding to the particular client. Therefore, the local entity may process the SAML request utilizing the action mapping provided in its partner list. The action mapping for example may map the permission to access particular records associated with the client on the partner entity to the local entity (e.g., payroll). Thus, the access permission established on the partner entity is mapped to a corresponding access permission on the local entity. Alternatively, the SAML request may pass the social security number of the client, as a field "ssn". The corresponding attribute mapping may then be utilized to map the "ssn" of the request to the "social security" namespace of the local entity. Thereafter the

local entity could determine access permissions based upon the content of the "social security" namespace of the client.

Furthermore, at step 640, a SAML assertion may be sent in response to the SAML request. The SAML assertion may be sent to the second entity according to the account mapping, attribute mapping, site attribute list, action mapping and/or the like. In an exemplary implementation, the local site may utilize a corresponding record in its partner list, such as a site attribute list, to determine what information to return in a SAML assertion. Therefore, the SAML assertion will contain the information required by the second entity.

Referring now to Figure 7, a block diagram of a system providing an interoperable single sign-on session, in accordance with one embodiment of the present invention, is shown. As depicted in Figure 7, the interoperable single sign-on system comprises a first entity 710 and a second entity 740. The first and second entities 710, 740 each comprise a session module 720, 750 and a partner list 730, 760, respectively. The session module 720 of the first entity 710 receives a SAML request 770 from the session module 750 of the second entity 740. The SAML request 770 contains an entity identifier, certificate, network address and/or the like. The session module 720 of the first entity 710 looks-up the entity identifier, certificate, network address and/or the like in its partner list 730. The record in the partner list 730 of the first entity 710 which contains the matching entity identifier, certificate, network address and/or the like includes an account mapping, an attribute mapping, a site attribute list, an action mapping and/or the like. Accordingly, the first entity 710 may map elements in the SAML request which relate to the content and organization of information

utilized by the second entity to the corresponding content and organization of information utilized by the first entity. The first entity 710 may also return a SAML assertion having elements which are required by the second entity 740.

The foregoing descriptions of specific embodiments of the present invention have been presented for purposes of illustration and description. They are not intended to be exhaustive or to limit the invention to the precise forms disclosed, and obviously many modifications and variations are possible in light of the above teaching. The embodiments were chosen and described in order to best explain the principles of the invention and its practical application, to thereby enable others skilled in the art to best utilize the invention and various embodiments with various modifications as are suited to the particular use contemplated. It is intended that the scope of the invention be defined by the Claims appended hereto and their equivalents.